International Journal of Advances in Scientific Research and Engineering (ijasre)

E-ISSN : 2454-8006

DOI: 10.31695/IJASRE.2024.7.1

Volume 10, Issue 7
July - 2024

A Proposed Block chain-Based Digital-Identity Management using ERC 725/735

Ammar A. Ali[1], Mustafa Q. Ali[2], Mustafa Hussein Zwayyer [3]

Assistance lecturer [1 2]

University of Baghdad

College of Islamic Sciences

Baghdad

Iraq

ABSTRACT

In modern technology, the ownership of electronic data is the key to securing their privacy and identity from any trace or interference. Therefore, a new identity management system called Digital Identity Management, implemented throughout recent years, acts as a holder of the identity data to maintain the holder's privacy and prevent identity theft. Therefore, an overwhelming number of users have two major problems, users who own data and third-party applications will handle it, and users who have no ownership of their data. Maintaining these identities will be a challenge these days. This paper proposes a system that solves the problem using blockchain technology for Digital Identity Management systems. Blockchain is a powerful technique to build a digital identity in chain matters that enables a secure environment. The idea of Blockchain is to distribute the data across multiple devices in a cryptographic way, which will reduce the ability to an impossible level. Therefore, in this paper a proposed Digital Identity based on Blockchain (ERC 725, and ERC 735) with MD6 as a hashing algorithm will be implemented in a Secure smart contract can prevent function calls from being carried out until the sender has received confirmation from a reliable issuer; for example, we might include a feature that restricts smart contract interactions to legitimate users only. Many additional use cases are possible with ERC-725, including multi-sig execution approvals and contract call verification in place of key validation.

Key Words: Block chain, ERC-725, ERC-735, Identity Management, Mechanism, Verification.

---

1. INTRODUCTION

A collection of characteristics associated with the entity, such as name, address, etc., can be used to describe the identity of an individual or an organization. Maintaining the data required for identity and controlling access to it are included in identity management. The three main players in the identity management system are the Holder, the Issuer, and the Verifier. [1,19]

Personal credentials can be issued for an identity holder (a legal individual or business) by the identity issuer, a reputable entity like the local government. The identity issuer certifies the accuracy of the personal data in that credential by releasing any user's data. The birthdate and last name, for instance. The identity holder may save such credentials in their personality identification wallet and utilize them at a later time to validate statements made by the identity data verifier, a third party. A credential is a compilation of several identity attributes, such as a name, age, and birthdate. An identity attribute is a piece of information about an identity. A verifiable claim that includes certain information about the bearer that is digitally signed and attested by the issuer is called a credential. Third parties that attest to the veracity of the data contained within credentials are the ones that issue them. The reliability and usefulness of a credential entirely hinge on the credibility and standing of the issuer. [1, 3, 20]

A credential's fact can be the holder's identity information (like their D.O.B.) or another kind of factual information (like their GPA). Anyone who has built a rapport of trust with the issuer—such as an employer—can serve as claim verification. The verifier asks for a particular credential (a person's birth certificate, for instance) and uses the issuer's signature to confirm the credential's authenticity. [2, 5]

Identity management is difficult if holders do not have complete control over their identity data. This is because the data are typically kept at the websites of third-party issuers, such as banks, credit agencies, and government agencies. These sites are thought to be the weakest link in the current identity management system because they are susceptible to data theft and hacking. As a result, the blockchain offers the potential to do away with middlemen while preserving citizens' autonomy over identity management. Holders are able to maintain control over the use of their identity data and ownership of their identities thanks to the notion of digital identity. [3,22]

The emphasize that a fundamental barrier to the widespread adoption of decentralized smart contracts is privacy, as many people and organizations view financial transactions—such as stock trading or insurance contracts—as extremely confidential. While certain privacy-preserving cryptocurrencies, such Zerocash and a few others, have made progress in their design, these systems do not allow for programmability, and it is not immediately evident how to provide programmability without disclosing transactions and data in plaintext to miners [4, 20].

In January 2009, Satoshi Nakamoto introduced two novel and unproven ideas at the same time when he started the Bitcoin blockchain. A decentralized cryptocurrency called "bitcoin" is the first.

Peer-to-peer digital currency that exists online and is not backed by a central issuer or has intrinsic value. The majority of public attention has been focused on "bitcoin" as a unit of currency thus far, due to its severe price volatility and the political ramifications of a currency without a central bank. [4]

But Satoshi's massive experiment also includes another, equally significant component: the idea of a blockchain based on proof of work that would enable consensus among the public over the sequence in which transactions should be made. [5]

When it comes to applications, Bitcoin may be characterized as a first-to-file system. This means that if an entity possesses 50 BTC and transmits the identical 50 BTC to both A and B at the same time, only the transaction that is confirmed first will be processed. The inability to intrinsically distinguish between two transactions that occurred earlier hindered the growth of decentralized digital currency for many years. One of the first reliable decentralized solutions was Satoshi's blockchain. And now, focus is quickly beginning to turn to this other aspect of Bitcoin technology and how the blockchain idea may be applied to purposes other than financial ones [5,21].

2. LITERATURE SURVEY

2008 saw the expression of the need for a more secure and dependable payment system, using cryptography-based proof of trust and doing away with the need for third parties, like banking institutions, to facilitate value transfers between interested parties directly. Satoshi Nakamoto may have been a pseudonym used by one or more anonymous individuals.

Since its launch in 2009, the open-source Bitcoin technology has enabled value transfers through the use of bitcoin cryptocurrency, which is created by the system itself. A public blockchain is used to manage and store transactions, and the technology aims to lower fees—such as those imposed by banks—and facilitate international negotiations.

The consensus process used by the nodes linked to the blockchain of Bitcoin, the first decentralized cryptocurrency, is based on proof-of-work and is governed by rules that specify how new blocks containing transactions will be added to the existing network. In a process known as cryptocurrency mining, these nodes—computational devices that are a part of the Bitcoin network, or the public blockchain that upholds it—must resolve a computational puzzle in order to be granted the authority to append a new transaction block to the existing blockchain. In exchange, they will receive bitcoins for their efforts [6].

The immutability of the data on the public blockchain, which is utilized to store bitcoin transactions, is another feature of the network that guarantees the accuracy of all the information kept there, offers a record of previous operations, and permits auditability and traceability. Furthermore, because Bitcoin is decentralized, its users can validate every transaction they make. In other words, the network as a whole can determine how many bitcoins each user account has received or sent by using a consensus-building mechanism that ensures the legitimacy of all activities [7].

Blockchains are systems that enable several functions; the most well-known example is Bitcoin.

The major focus of this work is on smart contracts, although these networks' adaptability extends beyond transactions involving values expressed in cryptocurrency. They enable order monitoring, decentralized real estate registration, and data storage, among other applications. In the realm of computers, every physical object can have its attributes, such as cost, color, weight, owner, etc., stated through software. This also applies to so-called intangible assets, like rights,

DOI: 10.31695/IJASRE.2024.7.1

private information, certificates, and trademarks, among others. Blockchain technology is used to store these kinds of digital assets in a secure and dependable manner, as well as suggest that the relationships between these assets may be facilitated by computer programming, or smart contracts, which function to automatically carry out certain commands in response to predetermined parameters and instructions. Though they bear the name "contract" in their title, smart contracts are not legally regarded as contracts; rather, they are merely a means of carrying out the terms stated in the actual contracts [8].

Nick Szabo first used the term "smart contracts" in 1994. According to this expert, smart contracts are computerized transaction protocols that carry out the conditions of the contract. According to some authors, smart contracts are software that was first created to automatically implement terms that two parties can agree upon when they sign a contract in an untrusted environment by utilizing the dependable computational capabilities of a blockchain network. Another way to define "smart contracts" is as computer programs that can be executed in a network of mutually trusted nodes, like the blockchain, without the need for a trusted authority to act as a middleman. Because these programs are resistant to manipulation, they can be useful in a variety of scenarios, particularly those that require money transfers in accordance with predetermined rules agreed upon by all parties, like financial services.

Smart contracts are digital contracts that lack the legality of traditional contracts and do not require the use of artificial intelligence resources. Instead, they enable the creation of clauses that are self-executing through automated execution and that are tamper-proof and dependent on a decentralized consensus. From a computing perspective, smart contracts are, to put it briefly, little programs that are designed to automate processes based on conditional "if" and "then" instructions. These programs are stored and executed in a decentralized manner in multiple devices connected in a peer-to-peer network, free from middlemen, and able to fulfil contractual clauses [9].

Within the realm of computing, smart contracts refer to software that carries out a logical series of actions based on specific clauses and regulations. In theory, smart contracts are composed of three components: the computational code that symbolizes the contract logic; the collection of messages that the contract can receive and that indicates the events that will trigger the contract; and the collection of functions that will initiate the reactions anticipated by the contract logic.

Regarding the many kinds of smart contracts, some subcategories are identified in the literature. For example, financial smart contracts are designed to facilitate transactions involving monetary assets. A portion of these agreements attest to the ownership, value, and supervision of conversations about tangible assets. Others with similar financial burden are designed for crowdsourcing, or collective finance, where they receive contributions from investors eager to support particular projects monetarily. The development of high-yield investment programs based on Ponzi (pyramid) schemes, which take money from interested parties and promise a return with interest on the amount invested as more interested parties join the project, has been another application for these smart contracts. Certain contracts offer protection against digitally provable events, like a certain flight being delayed at an airport, which would cause the beneficiary to receive a refund transfer. Other notarial smart contracts register ownership of assets and verify their provenance by utilizing the immutability of data found in blockchains. Some of these are utilized to store document hashes, guaranteeing the authenticity and consistency of these resources. This kind of smart contract is also used to link users' public keys to their true identities and to preserve copyrights in media such as music, artwork, and photos. Additionally, there exist digital wallet smart contracts that are designed to handle cryptographic keys, transmit transactions, and function as middlemen when interacting with blockchains. The last type is the library category, which consists of smart contracts whose features are carried out by other smart contracts. These functions can be used to convert text, values, and other formats [10].

A variety of blockchain implementations are available for the creation and maintenance of smart contracts. As previously said, Bitcoin is a system that primarily tries to transfer values using the cryptocurrency in a decentralized way utilizing a public blockchain that records all of the transactions that have taken place. Although a programming language with little resources can be used with Bitcoin's infrastructure, alternative protocols that enable the creation of simple smart contracts have been guided by the platform's open concept and immutability of data. Ethereum, a platform that utilizes a consensus method akin to Bitcoin and operates on its own blockchain, is a cryptocurrency known as Ether (ETH). Ethereum makes it possible to create smart contracts using programming languages like Solidity. Transactions sent to Ethereum's blockchain activate the contracts, and the network validates their results. Within the Ethereum universe, ether cryptocurrency can be sent and received between users and smart contracts. In

DOI: 10.31695/IJASRE.2024.7.1

addition to having its own blockchain, the stellar platform leverages a consensus algorithm based on the Byzantine Agreement Protocol, which reduces the number of resources needed to validate data blocks in relation to the cryptographic puzzle that Bitcoin needs to solve for mining and validation. [11]

Stellar, in contrast to Ethereum, does not have a proprietary language for writing smart contracts; instead, users can create these kinds of contracts by utilizing network transactions.

In addition to the previously listed, there is Lisk, which allows the development of private blockchains with customizable access permissions for outside parties and has its own coin. Smart contracts in Lisk can be written in JavaScript or Node.js and operate on other blockchains, while transactions utilizing the contracts can occur on the platform's main blockchain [12].

3. BLOCKCHAIN

Blockchain came to light following the widespread adoption of bitcoin. Although blockchain was first exclusively used for trade and financial transactions, a number of research have demonstrated that other applications of blockchain technology can be developed. This is a result of blockchain technology's high degree of transparency. For instance, with Bitcoin, the quantity of transactions and total amount of currencies in circulation can be easily and swiftly tracked due to the distributed structure of wallets. Because this system is peer-to-peer (P2P) based, it does not require central authority to approve or execute operations [13].

In this paper, the proposed method uses **MD6** instead of standard **SHA-256** hash algorithm as hash for previous and next hashing for each block, which will add more complexity to the security of hashing and reducing any duplicate to rare cases.

A blockchain is essentially a distributed, immutable, decentralized ledger composed of many blocks, each containing a collection of data. The blocks are joined using cryptographic techniques to produce a data chain that is chronological.

Data security is integrated into a blockchain's consensus process, which is a network of nodes that agree on a transaction's validity before adding it to the blockchain.

Blocks:

In a blockchain, a block consists of three primary parts as shown in Fig. 1:

1. The header includes metadata like the hash of the preceding block and a date with a random integer used in the mining process.
2. The primary and actual data, such as transactions and smart contracts that are saved in the block, are contained in the data portion.
3. Finally, for verification purposes, the hash is a distinct cryptographic value that serves as a representation of the complete block.

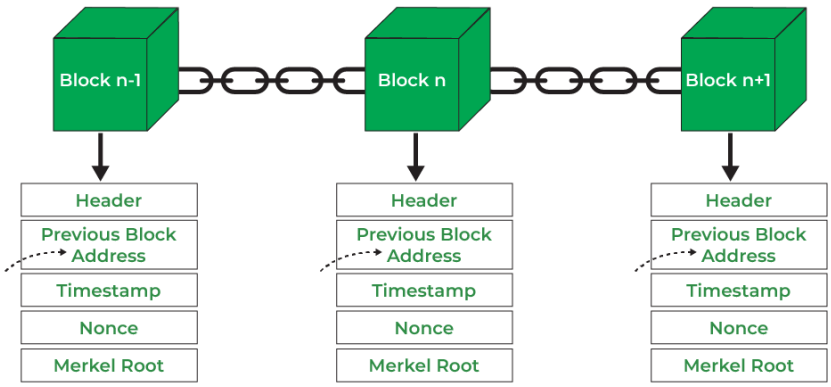

Figure 1. Blockchain Structure

Header: It serves as unique block identification for the whole blockchain. It manages every block in the blockchain. Miners routinely change the nonce value in a block header to hash it as part of their regular mining operation. The block header further contains three types of block information.

DOI: 10.31695/IJASRE.2024.7.1

Previous Hash/ Block Address: It connects the i+1th block to the ith block using the hash. It is, in essence, the hash of the chain's preceding (parent) block.

Timestamp: It is a mechanism that confirms the information in the block and gives digital documents a creation date or time. A string of characters known as the timestamp is used to identify a document or event uniquely and to show the date of creation.

Nonce: a nonce number with a single usage. It is an essential part of the evidence of work for the block. If the current aim is met or exceeded, It's contrasted with the actual goal. Those who, in order to ascertain the authenticity of a valued nonce, test, mine, and eliminate a significant number of nonces each second.

Merkel Root: It's a kind of data structure that organizes various data chunks. A Merkle Tree generates a digital fingerprint of each transaction, storing all of the transactions in a block. It enables users to confirm whether or not a transaction qualifies for inclusion in a block. [14].

4. SMART CONTRACT

The biggest innovation to support cryptocurrencies is Blockchain. There are several Blockchain iterations. The Ethereum Virtual Machine (EVM), commonly referred to as Ethereum Blockchain, enables a group of unidentified people to band together and collaborate under a virtual contract called a SC. Rules are necessary for contracts, however in this instance, the rules are included into the SC itself using a programming language called Solidity. Since SC lacks a "main" method, it cannot be executed by itself.

SC follows a deterministic path. When the SC is executed by any node in the Ethereum network, this constraint produces the identical result. Nodes can be either miners or users who must solve a mathematical puzzle in order to validate SCs. This validation procedure adheres to the consensus protocol, and the mathematical puzzle should provide the same result on all nodes.

After mining is finished, the owner uploads the SC to Blockchain. Miners discard contracts that do not align with Ethereum's regulations. This procedure can come after SC is resubmitted. As a result, the consensus process serves as a means of fostering mutual confidence amongst the participants.

It is believed that there are no mistakes or frauds. But the SC becomes unchangeable as soon as the owner uploads it. Thus, the trust can be destroyed by a dangerous SC. It could be abused by hackers, costing SC account holders a lot of money. Thus, before uploading SC to Blockchain, it is imperative to determine the SC's weaknesses [15].

The most well-known blockchain platforms, Ethereum and Bitcoin, use a process called proof-of-work (POW), which requires a significant amount of processing power from certain network nodes known as miners in order to solve a challenging computational problem and validate and add new blocks to the blockchain ledger. A miner's chances of winning the riddle increase with her computational power. All transactions that are legitimate in accordance with the blockchain's current state are accepted as blocks, and the winning miner publishes the puzzle's correct answer to other miners, the Smart Contract Structure as shown in Fig.2 [16].

DOI: 10.31695/IJASRE.2024.7.1

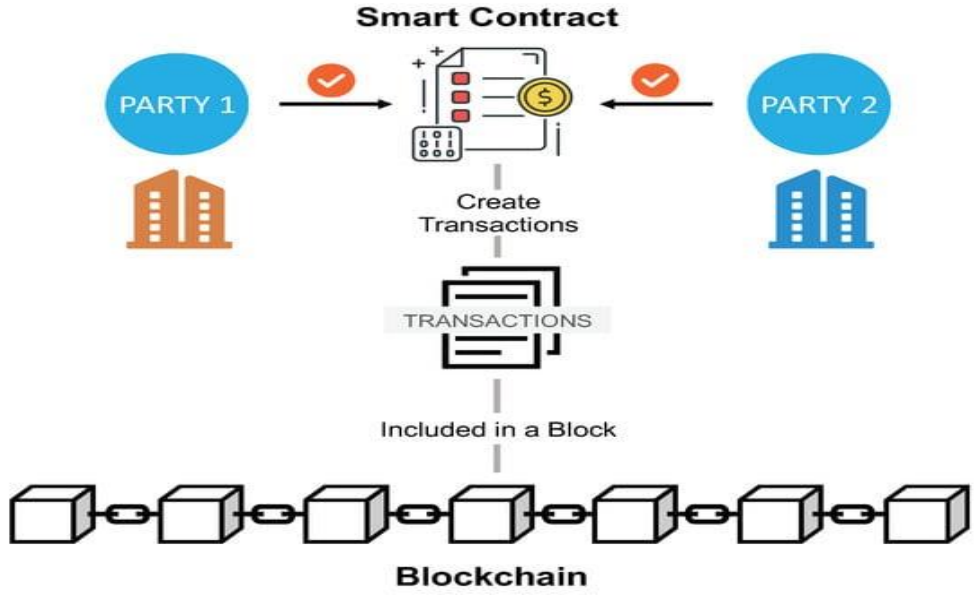

FIGURE 2. SMART CONTRACT STRUCTURE

5. IDENTITY

Identity is a collection of traits that characterize a person, thing, or group—abbreviated "Claim" for short. Features such as face, biography, background, and others could all be regarded as claims.

To ascertain an individual's, identify, the government and law enforcement agencies have released related documentation; a citizen's identity is a common example.

A citizen ID is documentation that you have visited the relevant authorities' offices and produced the necessary paperwork to prove your identification. You will receive a citizen ID from the authority confirming that you are the rightful owner of that identification in exchange.

Because so many unplanned events in life require citizen identification, users are fully aware of its significance. Many situations necessitate the identity verification process, such as obtaining a driver's license, purchasing a home, or opening an internet account [17].

6. PROPOSED METHOD

The proposed method is to implement Blockchain technology in Smart Contract using ERC 725 and ERC 735. The Ethereum Foundation developer Fabian Vogel Steller proposed the ERC-725 and ERC-735 standards for Ethereum smart contracts in 2017. An Ethereum blockchain standard interface for identity management is defined by ERC-725. In addition to enabling users to authenticate other users and sign transactions on their behalf, it gives users an interface to maintain and govern their own identity data. In order to give a consistent interface for maintaining identity claims on the Ethereum blockchain, ERC 735 expands upon ERC-725.

Fabian Vogel Stelle, the man behind Web3.js and ERC-20, has suggested ERC-725 as a unique standard for publishing and controlling an identity on the EVM-based blockchain. Other smart contracts and proxy smart contracts with multi-key control are described in the ERC-725 standard.

An ERC-725 identity smart contract can have its claims added or removed using the related ERC-735 standard. ERC-725 and ERC-735 aim to solve, among other important problems, the fact that consumers today do not actually control their data. When they are sleeping on the Internet, their identities are not adequately protected.

Numerous instances of user data leaks have been noted, encompassing everything from critical data like bank account numbers and passwords to more commonplace details like name, birthdate, and so forth.

The best solution to address this problem and help with identity management is blockchain. However, there aren't many emerging projects in this field because ERC-725 and ERC-735 are relatively new concepts.

The ERC-725 and ERC-735 Standards:

ERC-725 functions similarly to making accounts on the network; we may use these accounts to communicate on-chain without needing to store our private key or seed phrase. It yields several benefits once again:

DOI: 10.31695/IJASRE.2024.7.1

A. Make the user experience simple.

It will be unnecessary for users to commit seed phrases to memory when setting up a wallet for on-chain communication. Create these accounts so that users can enjoy a lot simpler experience than constantly backing up their seed phrases. These accounts are identical to the wallets that users use to run testnets, participate in airdrops, and communicate with other protocols.

Additionally, this will serve as the starting point for the widespread acceptance of cryptocurrency, which is finding a growing number of uses in fields like social networks with Social Token, gaming with the Play-to-Earn model, and DeFi.

B. Security.

Since the goal of ERC-725 is to give customers a more user-friendly experience with cryptocurrency, users can quickly replace the key for a more secure one for backup needs once they're happy with the results.

C. Establishing value accounts.

Account creation uses ERC-725, claim addition uses ERC-735. This implies:

- It is possible to add more wallet addresses with varying security settings and goals to a single account.
- Personal information can be added. Unlike the current wallet address, accounts with photos and biographies can communicate with other users on the chain.
- It is possible to incorporate statements that have been independently verified: It's akin to adding a LinkedIn bio.
- Utilize the systems for reputation.

And there's still a ton of untapped potential [18].

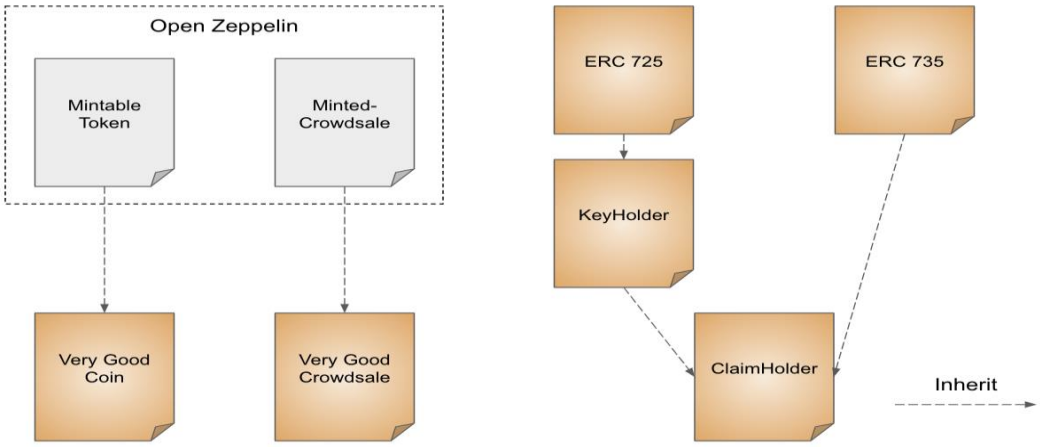

Figure 3. ERC 725/735

The implementation of the smart contract is very straightforward as shown in Fig.4:

DOI: 10.31695/IJASRE.2024.7.1

```
contract terms_condition_contract {
    /* Define variables and their types */
    bool buyerNegotiationState = false
    bool sellerNegotiationState = false
    bool dealState = false
    private string dealID

    FUNCTION initialization(string msg){
        Set dealID = dealID in msg
        IF sellerFlag != 1
            sellerNegotiationState = false
            Send msg to Seller that sellerTerms has been not been accepted and to modify
        ELSE
            sellerNegotiationState = true
        ENDIF
        IF buyerFlag != 1
            buyerNegotiationState = false
            Send msg to Buyer that buyerTerms has been not been accepted and to modify
        ELSE
            buyerNegotiationState = true
        ENDIF
        IF buyerNegotiationState = true AND sellerNegotiationState = true
            Set dealState = true
            Send msg to parent_deal_contract
        ELSE
            wait
        ENDIF
    }

    FUNCTION buyer_agreement_state(string msg) {
        Buyer Calls this FUNCTION to agree or disagree to Sellers Term
        FOR i = 1 to noOfTerms step 1 DO
            IF agreement in msg = sellerTerms[i]
                Set seller_Term_agreed[i] = true
                Send msg to Seller that sellerTerms[i] has been accepted
            ELSE
                Set seller_Term_agreed[i] = false
                Set sellerFlag = 1
                Send msg to Seller that sellerTerms[i] has been rejected
            ENDIF
        ENDFOR
    }

    FUNCTION seller_agreement_state(string msg) {
        Seller Calls this FUNCTION to agree or disagree to Buyers Term
        FOR i = 1 to noOfTerms step 1 DO
            IF agreement in msg = buyerTerms[i]
                Set buyer_Term_agreed[i] = true
                Send msg to Buyer that buyerTerms[i] has been accepted
            ELSE
                Set buyer_Term_agreed[i] = false
                Set buyerFlag = 1
                Send msg to Buyer that buyerTerms[i] has been accepted
            ENDIF
        ENDFOR

    }
}
```

Figure 4. The Implementation of the Smart Contract

Implementing Digital Identity using ERC 725/735 Implementing Digital Identity using ERC 725/735 Standards as shown in Fig.5:

Let's say we wish to sell airline tickets using a Listing contract, but we only want to engage with people whose email addresses have been confirmed. How can ERC 725 help us do this?

Let's first describe the entities that are going to interact:

• The person who wants to purchase the ticket is known as the Consumer.

• The identity that makes claims of the types "EMAIL_VERIFIED" and "PHONE_VERIFIED" is known as the Issuer. Only customers who have a verified email address and phone number from a reputable issuer will be permitted to use the listing.

DOI: 10.31695/IJASRE.2024.7.1

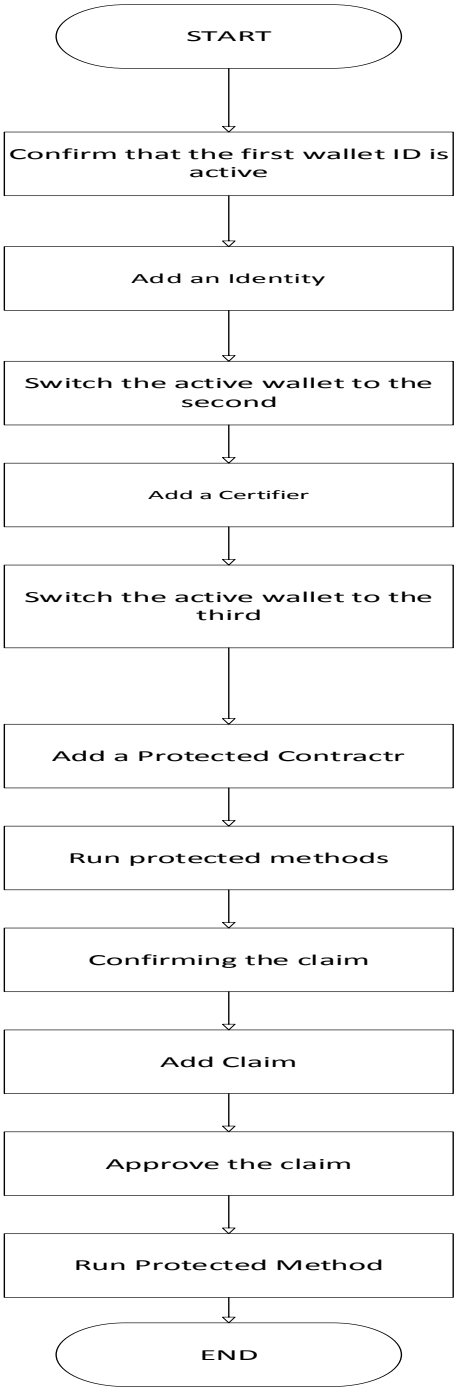

Figure 5. Proposed Method

This Raises A Few Questions:

1- How is an email address verified by the reliable Issuer?
2- How can a customer obtain a phone verified and email verified claim for their identity?
3- How can the Listing confirm that the Customer has a legitimate claim from a reliable Issuer that is both EMAIL_VERIFIED and PHONE_VERIFIED?

Let's walk through the process of establishing the necessary contracts and services, beginning with the Issuer, in order to respond to these queries.

Serving as a reliable third party is the Issuer's responsibility. Third parties will be able to trust trustworthy organizations in the future when they implement their own Issuer identification contracts onto the blockchain. Origin intends to provide their own fundamental Issuer contracts for the purpose of confirming phone numbers, email addresses, Twitter accounts, Facebook accounts, and so on. After that, other parties will be able to rely on the fact that these Origin Issuer contracts only make claims that are verifiably accurate.

DOI: 10.31695/IJASRE.2024.7.1

How does an email verifier operate? An application, such as http://example.com/verify-email, may be used in a conventional verification service. An email with a unique code is sent to the user, who then enters it back into the program. This will be a standard interface for email address verification in this application.
The email address can now be signed using a private key that is only known by the email verifier app once it has been validated. The issuer's identity has the matching public key. That's how a claim gets validated.

7. RESULTS

The performance of this proposed method with MD6 is showing that's high and efficient. The performance taken for hashing and the entire process is measured by the time taken to output the data. In this paper, the time monitored in Fig.6 and it's illustrated in Table 1:

Table 1. Proposed Method Performance with MD6

Hashing Algorithm	Time
SHA-256 Hashing	00:00:00.0021
MD6 Hashing	00:00:00.0016

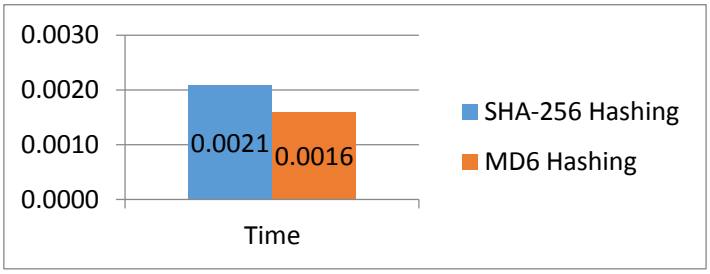

Figure 6. Proposed Method Performance with MD6

NIST standards shows very promising for the proposed hashing method, Table 2 shows the results of NIST for each hashing method:

Table 2. *NIST Standards for both Methods*

Standard	SH 256	MD 6
Runs Test	Non-Random	Random
Binary Matrix Rank Test	Random	Non-Random
Non-overlapping Template Match	Random	Random
Maurer's "Universal Statistical Test	Non-Random	Random
Serial Test	Non-Random	Random
Approximate Entropy Test	Non-Random	Random
Cumulative Sums Test (Forward)	Random	Non-Random
Frequency Test within a Block	Non-Random	Random
Test for the Longest Run of Ones	Random	Non-Random
Discrete Fourier Transform	Non-Random	Non-Random
Linear Complexity Test	Non-Random	Random

8. CONCLUSION

As blockchain technology advance quickly, developing smart contracts are becoming a popular area of study in both academia and business. Smart contracts' immutable and irreversible features can facilitate the transparent, conflict-free exchange of assets such as shares, cash, and intellectual property without involving a third party. Therefore, in the near future, smart contracts will be extensively utilized in social and financial institutions. We provide an overview of smart contracts in this paper, including their idea, implementation, and use cases. We also go over the difficulties the smart contract has and outline its potential developments. MD6 shows much higher performance and security than SHA-256, which can be recommended for blockchain applications. In the future, we intend to look at parallel blockchain technology and its uses for smart contracts.

DOI: 10.31695/IJASRE.2024.7.1

REFERENCES LIST

[1] Devi, S., Kotian, S., Kumavat, M., & Patel, D. (2022). Digital Identity Management System Using Blockchain. Available at SSRN 4127356.; doi:10.2139/ssrn.4127356

[2] Badr, A. M., Fourati, L. C., & Ayed, S. (2023). A Novel System for Confidential Medical Data Storage Using Chaskey Encryption and Blockchain Technology. Baghdad Science Journal, 20(6 (Suppl.)), 2651-2651. Available from: https://www.iasj.net/iasj/download/90f77a2f92ee1d58

[3] Phiri, J., & Agbinya, J. I. (2006, April). Modelling and information fusion in digital identity management systems. In *International Conference on Networking, International Conference on Systems and International Conference on Mobile Communications and Learning Technologies (ICNICONSMCL'06)* (pp. 181-181). IEEE. https://www.researchgate.net/publication/224633733

[4] Buterin, V. (2014). A next-generation smart contract and decentralized application platform. *white paper*, *3*(37), 2-1.

[5] Zhang D, Wahab A, Adi. Optimizing Blockchain Consensus: Incorporating Trust Value in the Practical Byzantine Fault Tolerance Algorithm with Boneh-Lynn-Shacham Aggregate Signature. Baghdad Science Journal. 2024 Feb 25;21(2(SI)):0633–3, doi: 10.21123/bsj.2024.9735

[6] Dixit, A., Asif, W., & Rajarajan, M. (2020, October). Smart-contract enabled decentralized identity management framework for industry 4.0. In *IECON 2020 The 46th Annual Conference of the IEEE Industrial Electronics Society* (pp. 2221-2227). IEEE.

[7] Mohammed NS, Dawood OA, Sagheer AM, Nafea AA. Secure Smart Contract Based on Blockchain to Prevent the Non-Repudiation Phenomenon. Baghdad Science Journal [Internet]. 2024 Jan 1 [cited 2024 Mar 8];21(1):0234–4. Available from: https://bsj.uobaghdad.edu.iq/index.php/BSJ/article/view/8164

[8] Kosba A, Miller A, Shi E, Wen Z, Papamanthou C. Hawk: The blockchain model of cryptography and privacy-preserving smart contracts. In: 2016 IEEE Symposium on Security and Privacy (SP). IEEE; 2016.

[9] Dixit, A., Asif, W., & Rajarajan, M. (2020, October). Smart-contract enabled decentralized identity management framework for industry 4.0. In *IECON 2020 The 46th Annual Conference of the IEEE Industrial Electronics Society* (pp. 2221-2227). IEEE, doi: http://dx.doi.org/10.21123/bsj.2022.7513

[10] Delmolino K, Arnett M, Kosba A, Miller A, Shi E. Step by step towards creating a safe smart contract: Lessons and insights from a cryptocurrency lab. In: Financial Cryptography and Data Security. Berlin, Heidelberg: Springer Berlin Heidelberg; 2016. p. 79–94.

[11] Bonneau J. EthIKS: Using ethereum to audit a CONIKS key transparency log. In: Financial Cryptography and Data Security. Berlin, Heidelberg: Springer Berlin Heidelberg; 2016. p. 95–105.

[12]Juels A, Kosba A, Shi E. The ring of gyges: Investigating the future of criminal smart contracts. In: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. New York, NY, USA: ACM; 2016.

[13]Bartoletti, M. Constant-deposit multiparty lotteries on Bitcoin. Cagliari, Italy: Universit`a degli Studi di Cagliari. https://en.bitcoin.it/wiki/Transaction_Malleability

[14] Zheng, Z., Xie, S., Dai, H., Chen, X., & Wang, H. (2017, June). An overview of blockchain technology: Architecture, consensus, and future trends. In *2017 IEEE international congress on big data (BigData congress)* (pp. 557-564). Ieee.. https://www.researchgate.net/publication/318131748

DOI: 10.31695/IJASRE.2024.7.1

[15] Chen, X. A systematic review of blockchain. Chengdu, China: Southwestern University of Finance and Economics. https://doi.org/10.1186/s40854-019-0147-z

[16]Zheng, Z. An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends. Guangzhou, China: School of Data and Computer Science, Sun Yat-sen University . https://ieeexplore.ieee.org/document/8029379

[17]Rossi Andrian, H. Blockchain Technology and Implementation : A Systematic Literature Review. bandung, Indonesia: 3School of Electrotical Engineering and Informatics, Institut Teknologi Bandung(ITB).

[18] Sung-Bong, J. A Survey of Blockchain and Its Applications. Gumi, Korea: Department Of IT-Convergence Kumoh National Institute Of Technology. https://ieeexplore.ieee.org/document/9219622

[19]Ali Khan, Z. Ethereum Smart Contracts: Vulnerabilities and their Classifications. Department of Computer Science Texas Tech University. https://ieeexplore.ieee.org/document/9439088

[20]Abuhashim, A. Smart Contract Designs on Blockchain Applications. Philadelphia, USA: Department of Computer and Information Science Temple University. https://ieeexplore.ieee.org/document/9439088

[21]Wang, S. An Overview of Smart Contract: Architecture, Applications, and Future Trends. Beijing 100190, China: The State Key Laboratory for Management and Control of Complex Systems, Institute of Automation, Chinese Academy of Sciences. https://ieeexplore.ieee.org/document/8500488

[22]ZenSar,. *Identity Management using Ethereum Blockchain Platform Enabling ERC 725 and ERC 735 Token for Identity Management*.

DOI: 10.31695/IJASRE.2024.7.1